UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/921,015	08/01/2001	Ronald Samuel Barchi	206004.00013	5803

22909 7590 04/13/2007

BANNER & WITCOFF, LTD.
1100 13th STREET, N.W.
SUITE 1200
WASHINGTON, DC 20005-4051

EXAMINER	
SYED, FARHAN M	
ART UNIT	PAPER NUMBER
2165	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	04/13/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	09/921,015	BARCHI ET AL.	
	Examiner	Art Unit	
	Farhan M. Syed	2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 March 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☐ Claim(s) *1,4,12-14,16-21,25 and 26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4,12-14,16-21,25 and 26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1, 4, 12-14, 16-19, 21-22, and 25-26 are pending.

2. The Examiner acknowledges the cancellation of claims 2, 3, 5-11, 15, 20 and 23-24 by the Applicant in the response to remarks/arguments submitted on 22 January 2007.

Continued Prosecution Application

3. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's Request of Continued Examination (RCE) submission filed on 22 January 2007 has been entered. In addition, the "Preliminary Amendment" filed 21 March 2007 has been entered for the continued examination of this application.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1, 4, 12-14, 16-19, 21-22, and 25-26 are rejected under 35 U.S.C. 102(e) as being anticipated by Ambrosini et al. (U.S. Patent No. 6,609,121 and Ambrosini hereinafter).

As to claims 1 and 14, Ambrosini teaches a method for processing calls (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) to a directory (i.e. "In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories.") (col 2, lines 16-67), comprising: receiving a call (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) to a directory, the call including one of a request (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) to add data to the directory (i.e. *"LDAP represents a simple, albeit powerful directory service which is capable of performing powerful directory service queries as well as allowing clients to issue commands that add, delete or modify directory service entries."*) (col 2, lines 16-67), a request to modify data in the directory (col 2, lines 16-67), or a request to delete data from the directory (col 2, lines 16-67), the call further includes at least one attribute (i.e. *" In performing a directory query, LDAP clients can choose filter attributes in the directory tree, for example a search location, and filter the search therefrom. Sample "base" values can include "st=FL . . . c=us" or "l=Boca Raton, l=Highland Beach Directory, st=FL, . . . , c=us"."*) (col 3, lines 48-53; col 4, lines 50-67; col 5, lines 1-67); evaluating (i.e. *"determining the schema rules that the entry must obey."*) (col 3, lines 15-18; col 5, lines 35-67) the attribute (Fig. 2; col 2, lines 16-67) according to a first rule (i.e. *"LDAP permits a user to control which attributes are required and*

allowed for a particular object class, thus determining the schema rules that the entry must obey." ... "Plug-in functions can be written to perform the following tasks: Validating data before the server performs an LDAP operation on the data;") (col 3, lines 15-18; col 5, lines 35-67) governing content of data that may be included in the directory (col 2, lines 16-67) and a second rule governing the structure of data that may be included in the directory (i.e. the validity standards/rules or schema rules; a set can consist of one or more entities in it.) (col 3, lines 15-18; col 5, lines 35-67); and processing the call (Fig. 2; col 2, lines 16-67) based upon the evaluation (i.e. *"determining the schema rules that the entry must obey."*) (col 3, lines 15-18; col 5, lines 35-67) of the call (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) according to the one or more previously determined rules (i.e. *"LDAP permits a user to control which attributes are required and allowed for a particular object class, thus determining the schema rules that the entry must obey." ... "Plug-in functions can be written to perform the following tasks: Validating data before the server performs an LDAP operation on the data;"*) (col 3, lines 15-18; col 5, lines 35-67) determining whether the attribute complies with one of the first rule and the second rule (i.e. determining if the request data/ entry is valid by some validity standards/rules) (col 3, lines 15-18; col 5, lines 35-67); forwarding (i.e. to send the client request to the directory if the validity of the client request is fulfilled) (col 3, lines 25-30; col 6, lines 35-47) the call to the directory (col 2, lines 16-67) when the call attribute complies with one of the first rule and the second rule (i.e. if the client request does not involve adding, deleting or modifying data then the request is for searching data and the searching request does not need validation and is merely forwarded/ sent to the directory) (col 3, lines 25-30; col 6, lines 35-47); and forwarding (i.e. to send the client request to the directory if the validity of the client request is fulfilled) (col 3, lines 25-30; col 6, lines 35-47) an error message (i.e. to send/ return an error message to the client) (col 6, lines 35-60) an error message to a source of the call (i.e. "For example, a

plug-in function can validate data before a new entry is added to the directory. If the data is invalid, the plug-in function can abort the LDAP add operation and return an error message to the LDAP client) (col 6, lines 35-48) when the call attribute does not comply with one of the first rule and the second rule (i.e. if the client request does not involve adding, deleting or modifying data then the request is for searching data and the searching request does not need validation and is merely forwarded/ sent to the directory) (col 3, lines 25-30; col 6, lines 35-47).

As to claim 4, Ambrosini teaches wherein the call (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) is forwarded to the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67) through a directory access server (i.e. the LDAP server) (col 5, lines 50-67) controlling access (i.e. *"For example, in order to restrict searches of a directory to entries exclusively including access control lists, the search phrase "objectclass=acl" can be specified so that only entries purporting to be access control lists are located."*) (col 3, lines 10-17) to the directory (col 2, lines 16-67).

As to claim 12, Ambrosini teaches that (i.e. *"determining the schema rules that the entry must obey."*) (col 3, lines 15-18; col 5, lines 35-67) determining whether (i.e. determining if the request data/ entry is valid by some validity standards/rules) (col 3, lines 15-18; col 5, lines 35-67) (i.e. *"In performing a directory query, LDAP clients can choose filter attributes in the directory tree, for example a search location, and filter the search therefrom. Sample "base" values can include "st=FL . . . c=us" or "l=Boca Raton, l=Highland Beach Directory, st=FL, . . . , c=us"*) (col 3, lines 48-53; col 4, lines 50-67; col 5, lines 1-67) the call attribute (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) complies (i.e. determining if the request data/ entry is valid by some validity

standards/rules) (col 3, lines 15-18; col 5, lines 35-67) with (i.e. the validity rule corresponding to the LDAP client's add operation that the plug-in function uses. A set can consist of one or more entities) (col 3, lines 15-18; col 5, lines 35-67) a data additional rule (i.e. the validity standards/rules or schema rules; a set can consist of one or more entities in it.) (col 3, lines 15-18; col 5, lines 35-67) when the call includes a request to add data (i.e. *"LDAP defines operations for interrogating and updating the directory. Furthermore, LDAP provides operations for adding and deleting an entry from the directory, changing an existing entry, and changing the name of an entry. Still the primary operation of LDAP is to search for information stored in the directory." ... "Further, if the LDAP server plug-in function is invoked before an LDAP operation executes, the plug-in function can prevent the LDAP operation from executing. For example, a plug-in function can validate data before a new entry is added to the directory."*) (col 3, lines 25-30; col 6, lines 35-47) to the directory, determining whether (col 3, lines 48-53; col 4, lines 50-67; col 5, lines 1-67) the call attribute (Fig. 2; col 2, lines 16-67) complies with a data modification rule (i.e. the validity rule corresponding to the LDAP client's modify operation that the plug-in function uses. A set can consist of one or more entities) (col 3, lines 15-18; col 5, lines 35-67) (i.e. the validity standards/rules or schema rules; a set can consist of one or more entities in it.) (col 3, lines 15-18; col 5, lines 35-67) when the call (Fig. 2; col 2, lines 16-67) includes a request to modify data (col 3, lines 25-30; col 6, lines 35-47) in the directory, and determining (col 3, lines 15-18; col 5, lines 35-67) whether (col 3, lines 48-53; col 4, lines 50-67; col 5, lines 1-67) the call attributes (Fig. 2; col 2, lines 16-67) complies with a data deletion rule (i.e. the validity rule corresponding to the LDAP client's delete operation that the plug-in function uses. A set can consist of one or more entities) (col 3, lines 15-18; col 5, lines 35-67) (i.e. the validity standards/rules or schema rules; a set can consist of one or more entities in it.) (col 3, lines 15-18; col 5, lines 35-67) when the call (Fig. 2; col 2, lines 16-67) includes a request to delete data (col 3, lines 25-30; col 6, lines 35-47) from the

directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67).

As to claim 13, Ambrosini teaches wherein the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67) employs the lightweight directory access protocol (i.e. LDAP) (col 2, lines 16-67).

As to claims 16 and 21, Ambrosini teaches wherein the attribute rule validator (i.e. plug-in function) (col 5, lines 35-67; col 6, lines 35-47) is capable of forwarding (i.e. The LDAP server calls the plug-in function before executing the LDAP operation (e.g. add, delete, modify or search); The plug-in function validates the operation and then sends the validation result to the LDAP server and the server executes the operation on the directory) (col 5, lines 35-67; col 6, lines 35-47) the call to the transaction monitor (i.e. the LDAP server) (col 5, lines 50-67), and the transaction monitor (i.e. the LDAP server) (col 5, lines 50-67) relays (col 5, lines 35-67; col 6, lines 35-47) is capable of relaying the call (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) to the directory through a directory access server that controls access to the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67).

As to claim 17 and 22, Ambrosini teaches wherein transaction monitor (i.e. the LDAP server (16), a software entity which is situated in the LDAP server) (Fig. 3; col 5, lines 50-67) is capable of relaying the call to the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67) through a directory access server

(i.e. the LDAP server) (col 5, lines 50-67) that controls access (i.e. *"For example, in order to restrict searches of a directory to entries exclusively including access control lists, the search phrase "objectclass=acl" can be specified so that only entries purporting to be access control lists are located."*) (col 3, lines 10-17) to the directory (col 2, lines 16-67).

As to claim 18, Ambrosini teaches wherein the rule validator (i.e. plug-in function) (col 5, lines 35-67; col 6, lines 35-47) is capable of forwarding (i.e. to send the client request to the directory if the validity of the client request is fulfilled) (col 3, lines 25-30; col 6, lines 35-47) the call (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) to the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67).

As to claim 19, Ambrosini teaches wherein the rule validator (i.e. plug-in function) (col 5, lines 35-67; col 6, lines 35-47) is capable of forwarding (i.e. The LDAP server calls the plug-in function before executing the LDAP operation (e.g. add, delete, modify or search); The plug-in function validates the operation and then sends the validation result to the LDAP server and the server executes the operation on the directory) (col 5, lines 35-67; col 6, lines 35-47) the call (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) to the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67) through a directory access server (i.e. the LDAP server) (col 5, lines 50-67) that controls access (i.e. *"For example, in order to restrict searches of a directory to entries exclusively including access control lists, the search phrase "objectclass=acl" can be specified so that only entries purporting to be access control lists are located."*) (col 3, lines 10-17) to the directory (col 2, lines 16-67).

As to claim 25, Ambrosini teaches a directory network (i.e. *"The LDAP directory service is based on a client-server model."*) (col 4, lines 15-17), including: one or more client computers (i.e. LDAP clients) (col 4, lines 15-120); a directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67), and an attribute rule enforcer (i.e. plug-in function) (col 5, lines 35-67; col 6, lines 35-47), the attribute rule enforcer (i.e. plug-in function) (col 5, lines 35-67; col 6, lines 35-47) being arranged in the directory network (col 4, lines 15-17) so as to intercept calls (i.e. client's query/ request into the directory database) (Fig. 2; col 2, lines 16-67) from the one or more client computers (i.e. LDAP clients) (col 4, lines 15-120) to the directory (col 2, lines 16-67).

As to claim 26, Ambrosini teaches the directory network (i.e. *"The LDAP directory service is based on a client-server model."*) (col 4, lines 15-17) further including directory access server (i.e. the LDAP server) (col 5, lines 50-67) is capable of controlling access (i.e. *"For example, in order to restrict searches of a directory to entries exclusively including access control lists, the search phrase "objectclass=acl" can be specified so that only entries purporting to be access control lists are located."*) (col 3, lines 10-17) to the directory (i.e. *"In LDAP, the basic unit of information consists of an entry. Entries are stored in a directories."*) (col 2, lines 16-67) interposed (i.e. works between the directory and the plug-in function) (col 5, lines 35-67; col 6, lines 35-47) between the attribute rule enforcer (i.e. plug-in function) (col 5, lines 35-67; col 6, lines 35-47) and the directory (col 2, lines 16-67).

Response to Remarks/Arguments

4. Applicant's arguments filed 22 January 2007 have been fully considered but they are not persuasive for the reasons set forth below.

Applicant argues:

(1) "However, the method for validating data disclosed in Ambrosini is very different from the method recited in claim 1 as amended herein. As disclosed in Ambrosini, the plug-in intercepts LDAP request to a directory and "converts the LDAP compatible search arguments to search arguments compatible with the DA [directory assistance] database..." Ambrosini at col. 7, lines 45-63 and col. 10, lines 40-60. Thus, the method disclosed in Ambrosini converts an entry whose data does not conform to acceptable data into an acceptable entry."

The Examiner respectfully disagrees with the Applicant. Converting the LDAP compatible search arguments to search arguments compatible with DA database is an additional layer that is being performed. *"Additionally, the plug-in can query the DA system* ***with the converted search arguments****, receive a result of listings therefrom in response to the query, convert the result set to a result set compatible with LDAP."* Column 7, lines 51-54. The preceding text clearly indicates that in order for a call to be forwarded to the directory, an additional conversion takes place for compatibility purposes and that the limitations recited in amended claim 1 is still being performed. That is, the converted search arguments encompass the limitations recited in claim 1.

This is akin to any request that is ***converted*** into binary form so that the request can be sent over a TCP connection from a source to the destination (e.g. the source is the client and the directory is the destination), because an ordinary person skilled in the art understands that communication between processors are converted into binary form so that they can be transmitted by the source to the destination, and when it is received by the destination, converted back into a compatible language structure to execute the request.

(2) "For example, among other recitations, there is no disclosure in Ambrosini of forwarding an error message to a source of the call when the attribute does not comply with the second rule governing the structure of data."

The Examiner respectfully disagrees. In the interview conducted between the Applicant and the Examiner on 21 March 2007, the Examiner understood that when the Applicant described the limitation of evaluating forwarding an error message to a source of the call when the attribute does not comply with the second rule governing the structure of data to mean that if the rules do not hold to an error message will generate and be sent back to the source. Similarly, the Examiner refers to the source code written in C programming language that illustrates an example method for issuing an LDAP command from a client to an LDAP server and receiving from the LDAPserver responsive directory information. See columns 3-6. The *ldap_perror (ld,"ldap_search_s")* function clearly indicates that an error message is generated and sent back to the source. An ordinary person skilled in the art understands that when

writing code that error validation schemes are used throughout the programming code that indicates where certain errors may occur.

Hence, the Applicant's arguments do not distinguish over the claimed invention over the prior art of record.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Farhan M. Syed whose telephone number is 571-272-7191. The examiner can normally be reached on 8:30AM-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey Gaffin can be reached on 571-272-4146. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

FMS

JEFFREY GAFFIN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100